(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
30 January 2003 (30.01.2003)

PCT

(10) International Publication Number
WO 03/009191 A2

(51) International Patent Classification[7]: G06F 17/60

(21) International Application Number: PCT/GB02/03291

(22) International Filing Date: 17 July 2002 (17.07.2002)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
01306245.0 20 July 2001 (20.07.2001) EP

(71) Applicant *(for all designated States except US)*: **BRITISH TELECOMMUNICATIONS PUBLIC LIMITED COMPANY** [GB/GB]; 81 Newgate Street, London EC1A 7AJ (GB).

(72) Inventors; and

(75) Inventors/Applicants *(for US only)*: **COLLINGRIDGE, Robert, John** [GB/GB]; 11 Dodson Vale, Kesgrave, Ipswich, Suffolk IP5 2GT (GB). **NEWBOULD, Richard, Eric** [GB/GB]; 8 Kelvin Road, Ipswich, Suffolk IP1 5EH (GB). **MARKWELL, Colin, Peter** [GB/GB]; 7 Kirby Close, Ipswich, Suffolk IP4 4PU (GB). **GOSLING, Timothy, David, Brendon** [GB/GB]; 205 Colchester Road, Ipswich, Suffolk IP4 4SL (GB). **ANDREWS, David, Joseph** [GB/GB]; 34 Frederick Square, Rotherhithe, London SE16 5XR (GB).

(74) Agent: **LLYOD, Barry, George, William**; BT Group Legal Services, Intellectual Property Department, Holborn Centre, 8th floor, 120 Holborn, London EC1N 2TE (GB).

(81) Designated States *(national)*: CA, US.

(84) Designated States *(regional)*: European patent (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, SK, TR).

Published:

— *with declaration under Article 17(2)(a); without abstract; title not checked by the International Searching Authority*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

WO 03/009191 A2

(54) Title: STORING AND ACCESSING PROFILE INFORMATION

(57) Abstract:

BNSDOCID: <WO_03009191A2_I_>

STORING AND ACCESSING PROFILE INFORMATION

The present invention relates to the storage and management of access to profile information, in particular to personal profile information. It has in recent years become increasingly common for computer systems to make use of personal profiles or personal data files to allow a user's interaction with a system to be personalised. In addition to storing personal information such as credit card details, telephone number, postal address and so on, a personal profile may include details of the user's personal preferences, for example to allow personalisation of a portal website with the user's specific interests.

The profile may be stored in a central network store, and may be accessed by a number of individual access providers/applications. The profile may be stored in such a way that it is platform independent, allowing access from a number of different devices, for example PC terminals, mobile telephones or portable or hand-held computer devices.

The present invention is a development of and an improvement to such profiles.

According to a first aspect of the invention there is provided a profile server system comprising: at least one profile server for storing a plurality of user profiles, each profile containing user-specific data and access control information; and an access controller which controls access to the user-specific data within a user profile as determined by the access control information stored within that profile.

In the present patent specification, a "user" is to be interpreted to include not only human entities, be they individuals or groups of users, but also computer and other types of entity that may be linked to the server system, having characteristics that may be represented in a profile. Similarly, references to "personal" information, e.g. "personal profiles", shall also be interpreted to include not only information of personal relevance to human users but also information

relevant to other types of entity such as computers.

The user specific data and access control information for each profile may be held in a common hierarchical data structure. Such a data structure may include such information as user interests, user defined bookmarks, user location, user preferences, client device details and so on. The user's specific data may also relate to one or more user roles, with at least some of the categories of information being replicated for each role. For example, there may be a telephone number field which is replicated for the "home" role and for the "work" role.

The role may be selected by the user/client, with the service provider being unaware of the role's existence.

By storing the user's specific information along with the access control information within an individual profile, the profile becomes portable and may be exported by its owner from one profile server to another. Each profile may be self contained, and the only thing that a service provider/application may need to know in order to use it is its location (i.e. the location of the profile server on which it is stored), along with suitable access credentials (e.g. a specific password for each service provider/application).

The user specific data within the profile may include weighting factors which decay with time. This approach may be particularly applicable to user preference information and interest information. An option may be provided to allow the user to inhibit the time decay of certain information.

To enhance the security of the data, the access controller preferably requires an accessing service (such as a service provider or application) to register a session and will allow access only to a single user profile within that session. If a service provider wishes to access a different user profile, it is preferably forced to supply new access credentials and to register again in another session. Where the profile includes role information, the access controller may permit access only to those user's specific data within the user profile which relate to that particular

user role. If the service provider wishes to access data relating to another role of the same user, it is preferably forced to re-register and to present new access credentials.

Profile servers according to the present invention are also compatible with known authentication and encryption mechanisms for use in communications between the profile server and service providers and between the profile server and users.

Preferably, the profile server stores a separate set of security credentials for each service provider within each individual profile. Where roles are used, further separate sets of security credentials may be stored for each role as well as for each user. Thus, the information within the profile server database is preferably not normalised.

In one embodiment, the profile server may analyse the user's specific data within a profile and may recommend or may automatically effect suitable updates to that profile. For example, the profile server may automatically analyse the user's bookmarks in order to recommend updates to the user interest information. The system could also analyse the incoming and outgoing e-mails for each user and recommend suitable updates to the interest information.

The profile may include current and/or historic user location information, for example forming a "location trail". This information may then be used either by the profile server or by an application accessing the profile server to predict where the user is going to be at some future time.

A user profile may be stored not only on the profile server but may also be cached or additionally stored as a duplicate on a local client device. This allows the user to have an anonymous personalised interaction with a service provider, for example by allowing the profile to disclose to the service provider only the interests information but not name and address information. With such an approach, the service provider is not even made aware of the location of the

profile server. The service provider needs to know only the IP address of the end user/client, and not the ID/password for the profile server.

Suitable synchronisation, either partial or total, may be provided between the central copy of the profile stored on the profile server and the user's local copy.

The access controller may be programmed to allow access not only to service providers/applications, but also to other locally or remotely stored profiles. This allows groups of users to set up "profile groups", with all of the profiles within that group automatically keeping themselves up to date by extracting, as necessary, information from other profiles within the group.

If a service provider/application requests access to data to which it is not authorised, the profile server may refer the request to the user and may allow access only if the user consents. The Service provider may allow the user four options: to refuse consent, to consent this time only, to consent only at specific times (e.g. 0900 to 1700, Monday to Friday) or to consent always (in order always to allow this particular Service Provider access to the data that it is now asking for).

The checking and setting of access rights, and the referral of requests back to the user where necessary, may be carried out by means of Session Initiation Protocol (SIP). Alternatively, HTTP and HTTP redirects may be used.

The profile server system of the present invention may preferably be used without the user having to access service providers/applications through a personalisation portal, nor should it be necessary for the user to have to logon to the profile server before accessing the service provider/application to be personalised. On the contrary, the user preferably logs straight into the website/application/service provider and either explicitly or by default informs the application of the profile server being used and the name or ID of the personal profile stored on it. The application itself then, without user intervention, may

approach the profile server and (assuming it is authorised) obtain and download the information it needs to personalise the application.

The invention extends to a profiling method using any of the features described above or in the specific description.

According to a further aspect of the present invention there is provided a method of personalising an application comprising:

(a) explicitly or by default informing the application of user profile information, including access details;

(b) requesting access to the profile, supplying the access details as well as application credentials;

(c) checking the access details and application credentials against access control information stored within the profile and, if authorised, allowing the application to access user-specific data within the profile; and

(d) personalising the application according to the user-specific data.

The invention may be carried into practice in a number of ways and one specific embodiment will now be described, by way of example, with reference to the drawings, in which:

Figure 1 shows in schematic form the main elements of a profile server system in accordance with the preferred embodiment of the invention;

Figure 2 illustrates the profile data structure;

Figure 3 illustrates the mechanism for granting access rights to a particular website;

Figure 4 shows a conventional prior art approach to the storage of access control information;

Figure 5 shows the way in which access control information is stored within the preferred embodiment;

Figure 6 illustrates message flows using HTTP redirects; and

Figure 7 illustrates message flows using Session Initiation Protocol (SIP).

The concept underlying the present invention, in its preferred embodiment, is that of a subscriber-based system with each subscriber having his or her own portable, platform-independent profile, stored on a central profile server. The information within the profile can be accessed by a variety of different applications being run by the user, provided that the user has granted appropriate access rights. That information may then be used automatically to personalise the application in some way, or to provide to the application confidential information such as credit card details.

Figure 1 illustrates schematically the main elements of a profile server according to the preferred embodiment of the invention. A profile server 10 uses an LDAP directory or relational database 12 to store a series of individual personal profiles 14. Access to the database is controlled by a gatekeeper 16 which, in this embodiment, uses Java beans and Java Server Pages (JSP).

Client devices 18 running user applications 20,22 may communicate with the gatekeeper 16 via the SOAP (Simple Object Access Protocol) or XML (Extensible Mark-up Language). Access may either be direct, as shown by the arrows 24, or may be via a SOAP SDK, as illustrated by reference numerals 26,28. A special PS Admin application 30 is provided to allow the user to manage his or her own personal profile 14.

The user applications 20,22 may be any application capable of talking to the profile server 10 which requires data to personalise the application. Examples include web pages, portal personalisation, internet bookmark management applications, games and so on. More generally, the applications 20,22 may represent service providers who need the information in the profile server to log the user on to their servers, to personalise it, or otherwise to obtain personal user information for use in one or more applications being provided by that service provider.

The applications may also be of the type which need personal information relating to the user, such as credit card information, addresses and so on. Instead

of the user having to type that information when requested by the application, the application instead simply obtains it (with reference to user permissions) from the profile server.

Communication with the gatekeeper 16 takes place using SOAP packets. Each SOAP packet may include several distinct blocks which themselves may represent distinct commands. Several commands may therefore be transmitted to the gatekeeper 16 within a single SOAP packet: for example, the first block in a packet might be a login command, the next block a read data command, the next block a write data command, and the last block a logout command.

More generally, the interaction between the applications 20,22 and the profile server 10 is session-based. There may be multiple sessions in a single request, and multiple blocks for a single session. The requests may be broken into blocks, and the blocks may be broken up into commands. Using a session ID, the system can tie subsequent blocks to commands used in previous blocks.

It is not essential for there to be a permanent on-line connection between the profile server 10 and the Client Devices 18. While the system is of course useful in such applications (for example using an internet connection), it can also be used for the provision of off-line services. In one example, a Client 18 which has access to the internet only via a dial-up connection may request by e-mail a list of local restaurants in London. An application 20,22 receiving such a request (which will include details of the location of the relevant user's profile server and an identifier for the respective profile) approaches the profile server to obtain information on the user's preferences. If the user has set up the profile to allow that particular application to access the information it needs, the profile server sends the information back to the application which e-mails the requested list of restaurants back to the user when the user next dials up.

The structure of the profile information 14, held within the Database 12, is shown in Figure 2.

Linked to a root 32 are Subscriber/User Object 34, of which there is one for each user. For each Subscriber 34 there is a Roles Container 36 having a number of individual Role Objects 38. Roles may be user defined and allow an individual a convenient way to store personal information which may vary between e.g. a work role and a home role.

Each Role contains personal information within a Personal Container 40 such as Credit Card Details 42, and Contact Information 44 including Postal Contact Details 46, Telecom Details 48 and Online Details 50. The user may also define for each role various interests, defined by Keywords 54 stored within an Interests Container 52. The context of the Keywords may be role-dependent, so that, for example, the Keyword "bus" may have a different context for a work role (in which it relates to computers) and a home role (in which it relates to a mode of transport). The Keywords may be provided with weightings, allowing the applications or service providers to access them, enhanced flexibility as to how those Keywords are to be used and combined.

Each role also has a Bookmarks Container 56 which holds a user-defined series of Folders 58,60 for keeping internet or other Bookmarks or Pointers.

A Preferences Container 62 includes areas where the user can set up individual preferences, on a role-by-role basis, for example for such things as Food 64 and Leisure 66. This allows Service Providers the opportunity to provide services such as the selection of local restaurants serving the user's preferred type of food, or the recommendation of places to visit when the user has some time to spare in an unfamiliar city.

A Services Container 68 holds Access Credentials 70 for each service provider/application that the user has permitted to access the profile. Under the Service Credentials are held Folders 72,74 which contain profile server Cookies.

A Dynamic Container 76 is used to store Client Information 78: typically, characterisation of client devices that wish to access the profile such as WAP

phone type, bandwidth available and so on. The information here may be populated by Service Providers, when given permission to do so by the user. Alternatively, the information here could be dynamically deduced from the characteristics of messages actually received by the profile server.

A Location Object 80 stores current and historic information on the location of the user, derived for example from User Manual Entry, GPS, cellphone ID and so on. This information may be used by a service provider to personalise its offerings to the user according to geographic location. The service provider could also use this information to predict where the user is going, and offer information to the user on that basis. The service provider could combine predicted location with preference or interest information: a user having an interest in pop music could be offered the opportunity to book a pop concert that is going to take place tomorrow in the town where the user will be staying. Or, for a user who has an interest in historic castles, the system could recommend a visit to a castle which is only a short detour from the user's journey.

Information contained in a user profile may also be used by service providers to generate an alert to a user. In order for a user to receive such an alert, the service provider may need to know not only static information about the user but also dynamic information, such as the user's current location. Such information may also be used to determine whether or not criteria defining an alert situation are currently met in respect of a particular user or group of users.

The entirety of the user's profile may be built up from a variety of different types of client interaction, for example interaction over a fixed line, interaction over a wireless or mobile link, local working and so on. The combination of all of these types of interaction will give a fuller overall picture of the user's behaviour. The end user therefore gets better personalised services due to more complete modelling of preferences and behaviour, either within the profile server itself or by the individual service providers on the basis of information provided by the profile server.

Control of the information contained within the profile is tightly maintained by the gatekeeper 16 (Figure 1), which allows or permits access to a particular requesting service provider according to the Credential Information 70 (Figure 2) which is held for that service provider. The default is to deny access: in other words, unless the user has specifically or impliedly granted rights, no access whatsoever is permitted.

Figure 3 illustrates schematically how a particular subscriber may grant access rights to a service provider or internet site X1. Details of X1, including the password required by X1 to access the profile server, are stored in the Credentials Object 70. This provides the information needed by the gatekeeper 16 either to permit or to deny X1 access to this particular subscriber's profile. Against each object within the profile, or against each specific field, is stored a tag indicating the type of access that the service provider X1 is permitted for that object or field. In the example shown in Figure 3, X1 has been granted both read and write access. Amongst other things, to the user's postal, telecom and online details. Read and write access has also been granted to the Client Information 78, allowing this service provider to update the profile with details of typical devices that may use its services, for example WAP phone details, bandwidth and so on.

Assume that, initially, service provider X1 does not have access rights to the subscriber's Credit Card Details 42. If the user wishes to purchase something from X1, and credit card details are required, X1 will first approach the profile server, asking for permission to have the credit card details released. In this example, the profile server will refuse. The user is then given the option either to type in the credit card details manually or, alternatively, to allow the profile server to release them to X1 either on a one-off or on a permanent basis. If a one-off release is authorised, the information is made available to X1 for this session only; if permanent access is granted, a Tag 86 is stored against the Credit Card Information 42 indicating that X1 has read-only rights to this information for this and for future sessions.

Roles could be handled either by the service provider or, transparently to

the service provider, by the profile server. In the former case, when the user logs onto a website, the site itself will need to request from the user details of the user's role as well as the address of the user's profile server. In the latter case, the service provider would know nothing about roles, and a request to the user to specify the role in use would then need to be generated automatically by the profile server either when the service provider logs on or when it requests access to specific information.

Stored within the profile is a Current Role Flag 82 (see Figure 2), which specifies the role that the user has currently selected. The service provider's access is role-limited, and a request from a service provider for information on a role other than the current role will be refused. Where the user is offline (and for example contactable only by dialup e-mail) the service provider's access is tied to a specific role, as specified in advance by the user. The service provider is not entitled to request information on any role other than the one to which it is tied.

Whenever a service provider asks the profile server for access rights, that can, where appropriate, be passed back to the user who is then given the opportunity to accept or deny the request. The user can specify (on a per service basis) that the access may be once only, never or always. If the last option is chosen, the user will no longer have to authenticate with that service provider each time he or she access the service, since that will be handled automatically by the profile server. This "breakout" mechanism acts to pass control from the service provider to the profile server, allowing the user to change the rights, and then reverting control back to the service provider.

More complex or sophisticated options could also be presented to the user. The user could for example allow access just for a limited period, or during certain recurring periods, such as every Monday or every third Wednesday. The user could also always allow access except at certain specified times, or on certain specified days. The permitted access could be at different levels, for example read, write, delete, create child nodes and so on. Combinations of these are also possible.

Figure 6 illustrates one possible security breakout mechanism for implementing this, using HTTP redirect. In the example shown, a Client 88 is accessing a service (for example a website) provided by a service provider 90. The user wishes to personalise the service using his own personal profile stored within his profile server 92.

First (but not shown) the Client 88 advises the service provider 90 that a particular profile service is to be used, and either explicitly or by default advises the service provider of the address of the profile server 92 to be used. In order to select a correct part of the profile, the user also has to specify, either explicitly or by default, which role is being used (e.g. home or work). That provides enough information for the service provider to instigate a session with the profile server and, provided that the rights have previously been granted to that Provider, to access some or all of the information. It should be understood that in a single session the service provider can access only a single user profile and, within that, a single role. There are separate security credentials for each service provider/user relationship, and for each role, so the service provider can access only a very constrained subset of the total information held by the profile server.

In the example shown in Figure 6, the service provider requires credit card details in order to complete a purchase, and sends a message (1) to the profile server requesting them. The service provider does not have access to those details, for this user in this role, and the profile server sends back a message (2) refusing the request. The service provider then uses an HTTP redirect (3) to pass the request to the profile server via the Client 88. Since the Client is authorised to obtain that information, the profile server responds with a further message (4) which is then passed on via HTTP redirect to the service provider. If and only if the client has authorised the service provider to have future access to this information, the service provider and the profile server may communicate directly, as shown by the messages (5).

An alternative security breakout mechanism, this time using Session

Initiation Protocol (SIP) is shown in Figure 7. Here, the service provider makes a request (1) of the profile server which the profile server has not been authorised to meet. Instead of refusing the request, the profile server sends a message (2) to the client asking whether the information can be released, and the client replies with a response (3). If the profile server has been authorised, it then replies to the service provider as shown at (4). Otherwise, it refuses the service provider's request.

The way in which the access control information is stored within the profile differs from standard approaches to the storage of such information. The standard approach – as shown in Figure 4 – would be to normalise the information in the Database 100 so that the Profile Data 104 is held in separate, linked, tables from the Access Control Information 101.

However, in the preferred embodiment, the Database 102 is not normalised, and the Access Control Information 103 is stored together with the profile information. This means that the access control information for a particular service provider SP1 may be duplicated within the database.

With a data-structure as shown in Figure 2, stored as shown in Figure 5, each personal profile is self-contained, independent and portable. The profile is stored in such a way providing for platform independence, and its portability means that it can be easily moved, should its owner so wish, from one Profile server to another. This can be done easily using a PS Admin Tool 30 (Figure 1) and does not require either the service providers or the profile server companies to share any knowledge or to communicate with each other in any way. It is therefore extremely easy for a user who is unhappy with the service provided by one Profile server company simply to transfer profiles to another.

The easy portability of individual profiles means that it is very easy for a company hosting a profile server service to host different profiles on different servers. Although a single server 12 is shown for simplicity in Figure 1, the profiles 14 could in fact be split across numerous servers, anywhere in the world.

Since each profile is independent and is complete in its own right, those multiple servers would not need to replicate their data, or to communicate with each other in any way. All that is required is that the Client can inform the service provider in some way of the location of the relevant user profile. In a large set-up, that could be conveniently achieved by maintaining a profile name server in a central location, that server maintaining a look-up index of user names or subscribers, cross-referenced to the address of the particular profile server on which that individual's profile is held.

In a variant of the embodiment already described, the user's information on the profile server 10 may also be stored or cached on the user's Client Device 18 – see reference numeral 10 in Figure 1. With this arrangement, service providers obtain their profile information not with reference to the remote profile server 10, but rather from the cached version 10 . This may be convenient for several reasons, not least in that it allows the user to have an anonymous personalised interaction with the service provider, without any need to disclose the user's profile server account details. The access control features allow the user to restrict the information seen by the service provider to any preferred subset of the whole information: the user might decide for example just to release details on interests but not name, address or other contact details. Anonymity may be enhanced through the use of IP addressing.

The local copy of the central profile could be a subset or a superset of the network-based profile. By tagging data that is network or locally based, the user can have control over which aspects of the profile are to be synchronised.

In a further variant of the invention, the information stored in the profile may be associated with weights which vary with time. Thus, the weighting associated with a particular interest or keyword could gradually decay over time if the user does not make any use of that interest. The entire history of the interaction that the user has with the profile may be recorded, and itself stored within the profile, so allowing service providers - where authorised – to have access to historical data. The use of that historical information may be controlled

by the service provider, using its own internal algorithms, or alternatively the profile server may itself apply suitable algorithms to the data to provide easily-digested "summary" information to the service providers. That "summary" information may itself be stored within the profile, of course with corresponding access control information. This provides an easy way for service providers to monitor for example whether a particular interest is periodic, or whether it is simply gradually decaying with time.

The interest information within the profile may be used by the profile server to drive knowledge management intelligent agents which search out relevant data from the internet. The information is presented to the user not necessarily immediately but when he or she can use it most effectively – for example when the user is at an appropriate location, at a convenient time, when the user next logs on and so on.

The profile may to a greater or lesser extent be self-populating. Service providers who have been given appropriate access can populate and update the profile automatically. The profile server could also proactively do this, and automatically seek out additional information for updating the profile. It could, for instance, analyse the user's bookmarks and suggest additional interests or keywords that could be added. The updating could either be done automatically or, preferably, with explicit user approval.

The following steps can be used to extract interest keywords from a set of bookmarks:

1. Retrieve a set of user bookmarks from the user's profile;
2. Fetch the content of the HTML pages which have been bookmarked, and the frames contained within them;
3. Process the text returned to find the keywords;
4. Present the keywords to the user; and
5. Add the user-selected keywords to the user's Keyword Store 54.

Keywords could also be extracted, in a similar way, from e-mail

correspondence to and from the user.

In a development of the invention, the profile server provides for the possibility of interaction between different user profiles, as permitted by the respective users. Thus, if appropriate consents are given, an individual profile may be accessed and/or modified not only by a service provider but by another profile. This allows groups of users to set up peer-to-peer links by means of which changes to one profile could automatically propagate through the profile of all the other users in the group. A simple example would be the use of self-updating address books: if one of the contacts within a user's address book shows a telephone number but no e-mail address, the e-mail address of that individual can automatically be obtained from the address book of one of the other users within the group. Once again, the updating of the user's profile could either be carried out automatically or only with the user's explicit consent on a case by case basis.

The following additional features, or any combination of them, may also be incorporated into the preferred embodiment:

- A user interface which allows queries/searches against information held within the database and/or external information.
- A function to alert service providers if changes have occurred to certain nodes. This could be access controlled, so the end user would need explicitly to enable this feature, either globally or on a node-by-node basis;
- The service provider may access the gatekeeper in either a stateless or a stateful fashion.
- Additional nodes/folders may be provided within the database structure to enhance the schema.
- A function to query for changes which have been made to data since a certain time. This may help in updating local caches.
- Batch requests may be supported (e.g. multiple commands in a single HTTP post) as well as single ones.
- Extra methods/functionality may be provided within certain areas of the database. For example, the interests area may allow the user to check the

relevancy of a particular article against a user/role combination using the keywords within that area.

5

BNSDOCID: <WO__03009191A2_I_>

CLAIMS:

1. A profile server system comprising: at least one profile server (10) for storing a plurality of user profiles, each profile containing user-specific data and access control information (103); and an access controller (16) which controls access to the user-specific data within a user profile as determined by the access control information stored within that profile.

2. A profile server system as claimed in claim 1 in which the user-specific data and the access control information (103) for each profile are held in a common hierarchical data structure.

3. A profile server system as claimed in claim 1 or claim 2 in which the profiles include role information (38) representative of user roles, at least some of the user-specific data being role-dependent.

4. A profile server system as claimed in any one of the preceding claims in which the user-specific data includes client data (78) representative of the characteristics of specific electronic devices (18) used by a user.

5. A profile server system as claimed in any one of the preceding claims in which the user-specific data includes user interest information (52).

6. A profile server system as claimed in any one of the preceding claims in which the user-specific data includes user-defined bookmarks (56).

7. A profile server system as claimed in any one of the preceding claims in which the user-specific data includes user-location information (80).

8. A profile server system as claimed in any one of the preceding claims in which the user-specific data includes user preference information (62).

9. A profile server system as claimed in any one of the preceding claims in

BNSDOCID: <WO_03009191A2_I_>

which the user-specific data includes weighting factors which decay with time.

10. A profile server system as claimed in any one of the preceding claims in which the access controller (16) requires an accessing service to register a session, and allows access to only a single user profile within a given session.

11. A profile server system as claimed in claim 10 when dependent upon claim 3 in which, during a given session, the access controller (16) permits access only to those user-specific data within a user profile which relate to a specific user role.

12. A profile server system as claimed in any one of claims 1 to 9 in which separate access control information (103) is stored, within each profile, for each service provider that has been authorised to access the profile.

13. A profile server system as claimed in any one of the preceding claims in which the profile server (10) analyses the user-specific data within a profile and recommends or effects suitable updates to that profile.

14. A profile server system as claimed in claim 13 when dependent upon claims 5 and 6, in which the profile server (10) analyses the bookmarks (56) and recommends or effects suitable updates to the user interest information (52).

15. A profile server system as claimed in claim 7 in which the profile server (10) determines a probable destination location from historic user-location information (80).

16. A profile server system as claimed in any one of the preceding claims in which a given user profile is also cached or stored as a duplicate (10) on a local client device (18) operated by an owner of the profile.

17. A profile server system as claimed in any one of the preceding claims in which the access control information within a profile authorises the access controller (16) to allow access to other locally or remotely stored profiles, or to

BNSDOCID: <WO__03009191A2_I_>

users who own such other profiles.

18. A profile server system as claimed in any one of the preceding claims including a personalisable application which, when informed by a user of the location of the profile server (10) which stores that user's profile, obtains at least some user-specific data from the said profile and personalises itself accordingly.

19. A profile server system as claimed in claim 16 including a personalisable application which obtains at least some user-specific data from the said cached or duplicate copy (10) in the local client device (18) and personalises itself accordingly.

20. A profile server system as claimed in claim 18 in which, if the application requests access to data to which it is not authorised, the profile server (10) refers the request to the user and allows access only if the user consents.

21. A profile server system as claimed in claim 20 in which the profile server (10) provides options allowing the user to decline consent, to consent this time only, and to consent always for this application.

22. A profile server system as claimed in claim 20 or claim 21 in which the referral of the request uses HTTP redirect.

23. A profile server system as claimed in claim 20 or claim 21 in which the referral of the request uses Session Initiation Protocol (SIP).

24. A profile server system as claimed in claim 7 in which the user-location information includes past locations.

25. A profile server system as claimed in claim 7 or claim 24 in which the user-location information includes a predicted future position.

26. A profile server system as in Claim 1, wherein said user is a computer.

27. A method of personalising an application comprising:

(a) explicitly or by default informing the application of user profile information, including access details;

(b) requesting access to the profile, supplying the access details as well as application credentials;

(c) checking the access details and application credentials against access control information (103) stored within the profile and, if authorised, allowing the application to access user-specific data within the profile; and

(d) personalising the application according to the user-specific data.

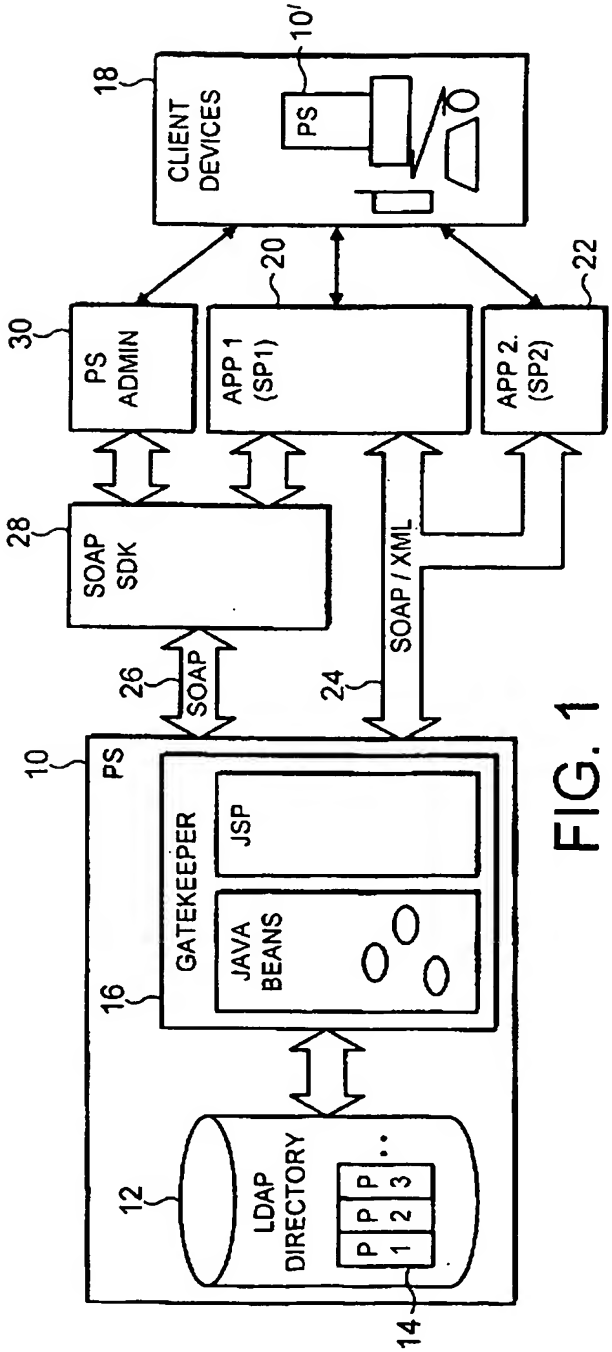
CLIENT DEVICES
PS
10'
18
20
22
PS ADMIN
APP 1 (SP1)
APP 2. (SP2)
30
SOAP SDK
28
SOAP
26
SOAP / XML
24
10
PS
GATEKEEPER
JSP
JAVA BEANS
16
LDAP DIRECTORY
P 1
P 2
P 3
12
14

FIG. 1

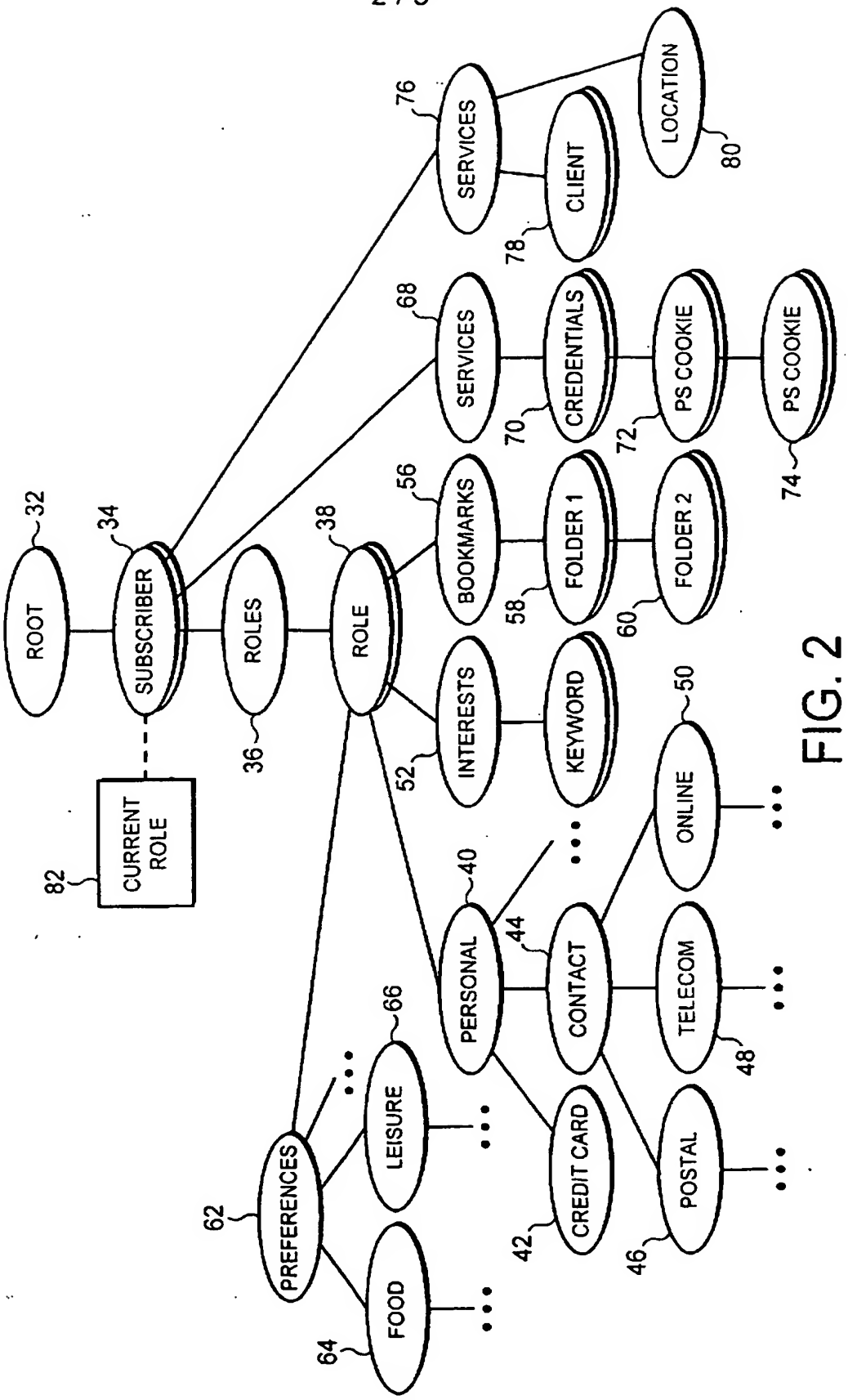

FIG. 2

BNSDOCID: <WO_03009191A2_I_>

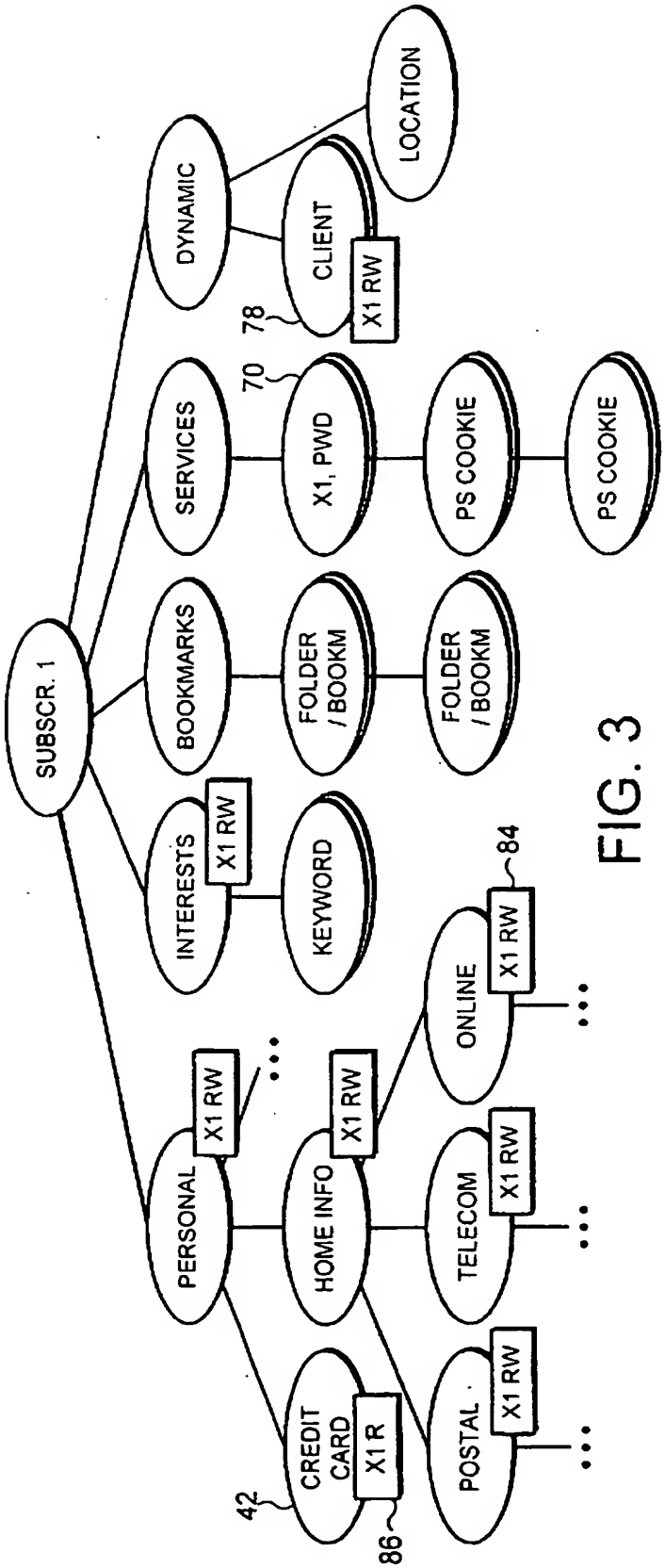
SUBSCR. 1
DYNAMIC
LOCATION
CLIENT
X1 RW
78
SERVICES
70
X1, PWD
PS COOKIE
PS COOKIE
BOOKMARKS
FOLDER / BOOKM
FOLDER / BOOKM
INTERESTS
X1 RW
KEYWORD
PERSONAL
X1 RW
HOME INFO
X1 RW
ONLINE
X1 RW
84
TELECOM
X1 RW
CREDIT CARD
X1 R
42
86
POSTAL
X1 RW

FIG. 3

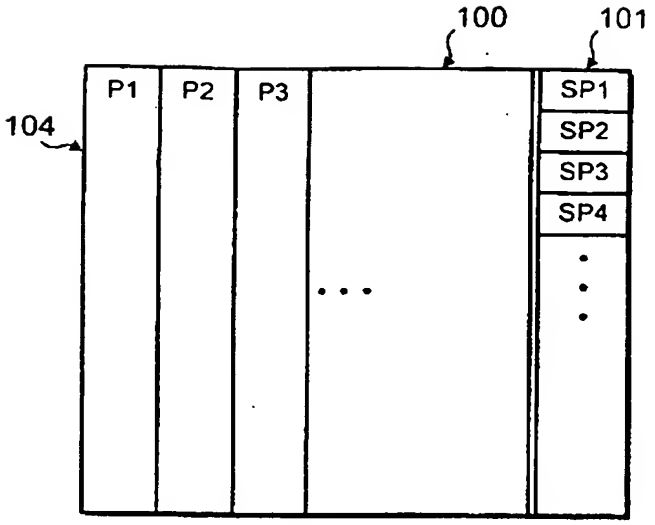
100
101
104
P1
P2
P3
SP1
SP2
SP3
SP4

FIG. 4

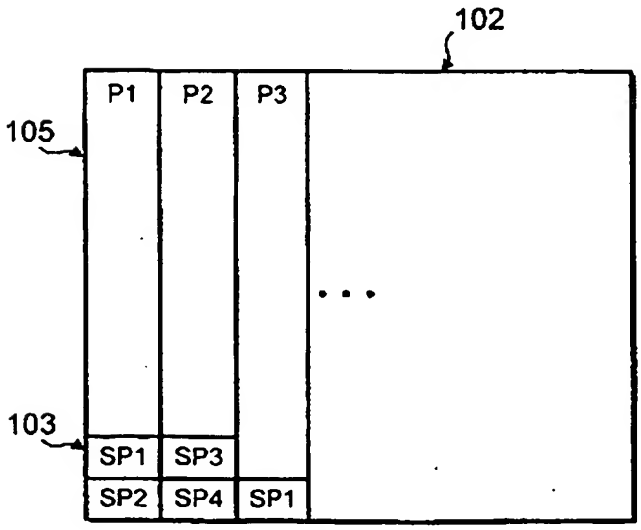
102
105
103
P1
P2
P3
SP1
SP3
SP2
SP4
SP1

FIG. 5

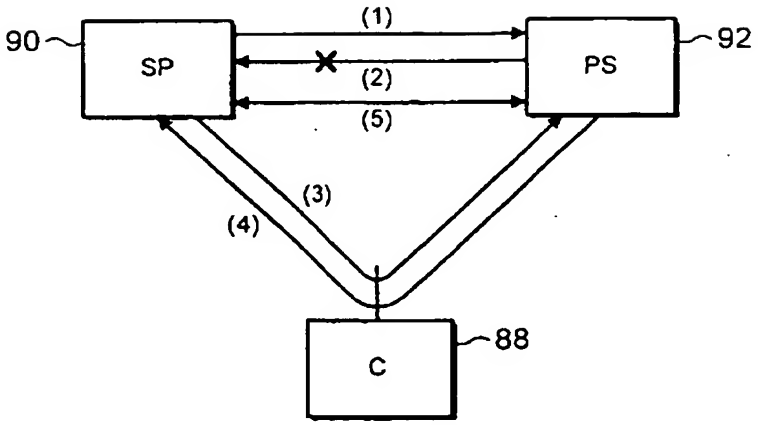

FIG. 6

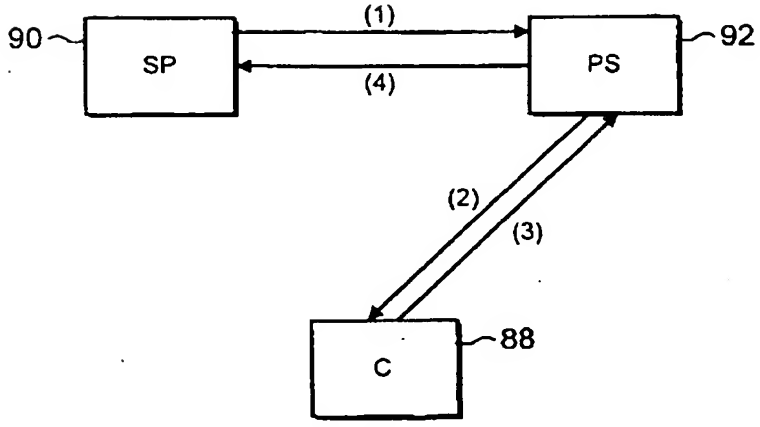

FIG. 7